International Journal of Computer Networking, Wireless and Mobile Communications (IJCNWMC)
ISSN(P): 2250-1568; ISSN(E): 2278-9448
Vol. 5, Issue 1, Feb 2015, 13-18
© TJPRC Pvt. Ltd.

TRANS STELLAR
• Journal Publications • Research Consultancy

MICRO UTILITIES COMMUNICATION UTILITIES ON LOW-COST MOBILES

GOPAKUMARAN T. THAMPI[1] & JAYALAKSHMI SRINIVASAN[2]

[1]Principal, Thadomal Shahani Engineering College, Mumbai, Maharashtra, India

[2]Assistant Professor, Department of B.Sc.,IT, V. E. S. College of Arts, Science & Commerce, Mumbai, Maharashtra, India

ABSTRACT

With abrupt advancement in mobile technologies more people are connecting to each other in different ways. There was a time when the applications were only available in the ordinary desktop computers. But now, it is available also in wireless devices taking the current state of the technology available for accessing 3G networks into consideration. Mobile networks evolve into a more powerful medium for information sharing and the devices that are connected to them grow in their ability to process the data at faster rates. Chat applications are built-in with only some brand of mobile phones and they are compatible with them alone. They are costly and difficult for the naive ones to procure.

As per the recent statistics, out of 1 billion mobile users in India only 137 million users are using internet. So to **serve the rest of the users**, designing mobile application with necessary utilities is mandatory. This paper "Micro Utilities" is an attempt to bring the overall effectiveness of mobile applications with Micro Mail, Micro SMS and chatting via Bluetooth in a cost-effective manner to satisfy all types of mobile users. The first phase of the research deals with Micro Mail. During this phase of the research, the author proposes that using this Micro Mail utility, Users can send email as the mobile is configured with the SMTP port of the Micro Mail.

This work is concentrated on improving the user experience with the utility along with the ability to send emails. The second phase of the research deals with Bluetooth chats. During this phase of the research, the author proposes that the users wanting to have a secure private chat within the radius of 20 meters can chat using this utility. This work is concentrated on providing cost-effective low distance secure communication amongst the users. The third phase of the research deals with sending micro SMS to other mobile users. This phase ensures the customized User Interface screen for sending the SMS. This paper in overall suggests a model for mobile users in terms of communication utility there by providing a less economical facility for low cost mobiles.

KEYWORDS: Micro Utilities, Bluetooth Chat, Micro Mail, Micro SMS

INTRODUCTION

Cell phones have become a part of our daily lives. They have evolved to such a point now, that people cannot think of a life without mobiles. Now days, The cell phone users can take pictures, play games and most importantly send, receive and store data. These are possible because of the internet connections in the mobiles which ensure the better connectivity and competencies of the users. As per the census report of 2014[1], out of 1.27 billion population, 137 million users are using internet in mobiles. They contribute only 10.84% to the total population. This paper is to concentrate the left out 89.16% of Indian populations by providing cost-effective and the most useful utilities by using J2ME technologies. Improving the user experience of low-cost mobile users by introducing the new utilities for communication such as sending e-mail, SMS and Chat via Bluetooth is the main objective of this paper. The research is limited to only those utilities as

specified in objectives and it targets the audience group having low cost mobiles and wanting to have basic Smartphone like utilities.

TECHNOLOGY SELECTION

The researcher selects J2ME as the technology to build the utilities.

The **J2ME** is a specification of a subset of the Java platform aimed at providing a certified collection of Java APIs for the development of software for small, resource-constrained devices. The **CLDC** (Connected Limited Device Configuration) contains a strict subset of the Java class libraries, and is the minimal needed for a Java virtual machine to operate. CLDC is basically used to classify countless devices into a fixed configuration. When coupled with one or more profiles, the Connected Limited Device Configuration gives developers a solid Java platform for creating applications for consumer and embedded devices. **MIDP** (Mobile Information Device Profile) boasts GUI API, and MIDP 2.0 includes a basic 2D gaming API. Applications written for this profile are called MIDlets. Almost all new cell phones come with a MIDP implementation.

J2ME PROFILE

Fengetal (2001) has stated that a profile is "Built on top of a specific configuration, a J2ME profile defines class libraries to address the specific demands of certain market segment." In other words a profile addresses a specific class of device, such as pagers and mobile phones. It supplements the already existing functionality in a configuration for these devices by sitting on top of the configuration.

There are currently two profiles: the **Foundation Profile (FP)** which supplements the CLDC, and the **Mobile Information Device Profile (MIDP)** which supplements the **CLDC**. CLDC is the smaller of the two J2ME configurations, aimed at serving devices that have the following characteristics:

- 16-bit or 32-bit processor
- Limited memory – 512 KB or less
- A limited power supply (typically a battery)
- Limited or intermittent network connectivity
- A simple user interface, if any at all
- Limited screen size
- Limited input abilities

CLDC is concerned with applications destined for use on mobile phones, pagers, Personal Digital Assistants (PDAs), and similar devices. In order for the CLDC to fit onto these devices with such memory limitations much of the standard functionality present in the Java 2 Standard Edition (J2SE) is abandoned. For example, the java.util.package, which contains forty-seven classes and interfaces in the J2SE, was reduced to only ten classes in the CLDC. The result is that CLDC does not provide all of the functionality required to build useful applications (E.g. user interface classes). But this was Sun's intention, as CLDC was not intended to be a complete solution; it is a common base upon which profiles targeted at specific product types can add functionality too.

MIDP

The Mobile Information Device Profile (MIDP) adds to the CLDC the functionality necessary to build useful applications for mobile phones, two-way pagers and similar devices. MIDP re-implements functionality to support user interfaces, timers, simple persistent storage, networking and messaging. Applications developed to the MIDP specification are called **Midlets.** An application developed for use on a MIDP device is referred to as a MIDlet. A MIDlet must also define a public noargument constructor which is the method instantiated by the **Application Management Software (AMS)** on the MIDP device when a MIDlet is launched. In order to develop an application to the MIDP/CLDC specifications, the MIDP and CLDC class packages must be downloaded [JavaSun J2ME, 2001] and their locations added to the classpath in the IDE where the MIDlet is being developed. All the class files belonging to a MIDlet are packaged into a single JAR file. This can then be downloaded and installed onto a wireless device via a serial cable connected to a PC or over-the-air (OTA) via a wireless network. This JAR file must also include a **manifest** file.

To aid with the packaging of MIDlets Sun provide the J2ME Wireless Toolkit (J2MEWTK) [JavaSun J2ME, 2001]. With this toolkit one can streamline several of the necessary tasks (creation of manifest file and application descriptor, compilation using MIDP compiler, and precertification) into one easy step. This facility is provided by a part of the toolkit called the Ktoolbar. Alternatively the J2MEWTK can be integrated with the 'Forte for Java' IDE. These features are subsequently accessible via 'Forte for Java' instead. The User Interface defined in MIDP is logically composed of two sets of APIs: -

- **High-level UI API** which emphasizes portability across different devices –
- **Low-level UI API** which emphasizes flexibility and control.

The portability in the **high-level API** is achieved by employing a high level of abstraction. The actual drawing and processing user interactions are performed by implementations.

Applications that use the high-level API have little control over the visual appearance of components, and can only access high-level UI events. On the other hand, using the **low-level API**, an application has full control of appearance, and can directly access input devices and handle primitive events generated by user interaction.

However the low-level API may be device-dependent, so applications developed using it will not be portable to other devices with a varying screen size.

MICRO MAIL

In this utility, the mobile phone is configured with the SMTP port of Gmail so that the users can send email which is the most important utility the Smartphone users enjoy. We don't always sit in front of desktop and wait for important emails and so by syncing the email with the mobile, we can keep our mails up to date. The researcher proposes to use Java Mail API to achieve the objectives of this utility.

This utility involves the following processes.

- User Input
- File storage

- Read from file
- Send the mail
- Acknowledgement message

The proposed work accepts Username, Password, Subject, Recipient addresses and the message to be sent as inputs through the emulator, user interface screen provided by Netbeans IDE.

The second functionality involves the creation of XML file to store the user-input through javax.servlet.iostream package.

The third functionality reads the xml file for the further processing of data. To transform the information it uses javax.xml.parsers package.

The next functionality, which is vital, sends the mail to the specified recipient(s). To achieve this, it uses Java Mail API and its objects for transport, session, send and receive messages. To send the mail, Researcher used the package javax.mail.internet.

The Java Servlet with IOStream ensures the input parameters are processed appropriately by using Java Mail API. The arguments are transmitted by using XML as carrier.

The final functionality will prompt a response message after the mail has been sent successfully. Thus it generates acknowledgement signal. This is achieved with the help of javax.mail.internet package.

BLUETOOTH CHAT

Bluetooth is a technology which provides direct connection to the digital appliances around you.

It eliminates the need of inconvenient cable attachments that is it will enable users to connect a wide range of devices simply, without the need of connecting cables. It automatically connects to the previously connected devices. Thus, it makes wireless communication and networking between devices easy.

It enables the usage of mobile data in a variety of applications. In Bluetooth all the connections between devices are instant and invisible and the devices can talk even if they are not in line of sight because Bluetooth utilizes a radio-based link.

The author proposes the chat application using Bluetooth as it offers the following advantages:

- Chatting in more secured manner-with no fear of backup server
- Chatting even when no line of sight-with no worries of network range
- Simple and powerful communication -such as your car mechanic send a message to your mobile phone telling you that your car is ready when you are in service center
- Effective communication-such as sharing of the classified message during office meetings or during war at border
- It ensures connective convenience-easily getting connected with any device such as phones, laptops. PDAs and mobile phones

This utility involves the following phases:

- User Input
- Device Discoverer
- Chat Controller
- Service Discoverer

The proposed work accepts the micro chat as an option for initiating chatting through the emulator, user interface screen provided by Net beans IDE.

The second functionality involves the device discoverer that discovers the nearby devices so as to find members for chatting. The researcher proposes the use of javax.bluetooth package for the same.

The third functionality works upon acceptance of the members. The functionality is achieved through the concept of Service Discoverer to ensure the prompt delivery of the messages. javax.bluetooth package has necessary methods to achieve the same.

The next functionality, which is vital, controls and acts as the server for the messages, senders and receivers. The researcher proposes the creation of Chat controller to do the major server tasks by importing javax.microedition package and javax.bluetooth package.

MICRO SMS

In this utility, the researcher proposes sending of SMS to other mobile users in a customised environment by offering user-interface screens to the users.

The researcher proposes the usage of wireless messaging API (WMA) to accomplish the same. Wireless messaging API sits on CLDC so as to send SMS and CBS. It supports 160 characters of SMS and the evolution of WMA, WMA 2.0 enhances, MMS. SMS and MMS use stored network for broadcasting the messages.

This utility involves the following processes.

- User Input
- Send the message
- Acknowledgement message

The proposed work accepts Option of a micro SMS service, Recipient Phone number and the message to be sent as inputs through the emulator, user interface screen provided by Netbeans IDE.

The second functionality involves the vital task of sending SMS to the specified phone number. In order to ensure the integrity, the author proposes the usage of javax.microedition and javax.wireless.messaging packages.

The final functionality will prompt a response message after the message has been sent successfully. Thus it generates acknowledgement signal. This is achieved with the help of javax.wireless.messaging package.

CONCLUSIONS

The main objective of this research is to provide several utilities to the low-cost mobile users in less economical way. By providing the above specified 3 micro utilities, the objective is fulfilled to the extent of its satisfaction. As Java is freeware, the proposed software is free- of- cost. So its primary focus on user satisfaction is met. A good cause, in terms of micro utilities, by keeping user's perspectives in mind, this research gains competitive advantage. At the same time, this research keeps 90% (approx.) of the mobile users in India who are not having internet in their mobiles. Even if they are affordable to procure the connection for the first time, the maintenance cost is the major hurdle for them for not to be in the smart phone users' crowd. The mind-set and the mentality add to that hurdle.

The research indirectly aims on "Social Responsibility" factor. If the utilities are used in their low-cost mobiles, the desire to go for the new phones just to enjoy these utilities will be reduced. M-Waste and the hazardous it creates to the society will also be reduced.

In spite of some of the advantages, the research witnesses some of the limitations such as technological constraints. As J2ME has limitations with database, data storage and its persistent, the research also has its limitations when it has to implemented large scale. The future prospects of J2ME and its compatibility to support further versions, attaining large economies of scale is not far from reality.

REFERENCES

1. Paper of Puneet Gupta, a technology specialist in Radio networks and freelance technology writer based in Bangalore, India.
2. MobileInfo.Com technical staff.
3. www.infm.ulst.ac.uk
4. www.r4r.co.in
5. www.bsria.co.uk
6. Paper submitted to Unitex Information Technology on Bluetooth
7. home.graffiti.net
8. Paper submitted to Informatics Education Limited on Wireless Messaging